

FOSSpicks

Sparkling gems and new releases from the world of Free and Open Source Software



Mike Saunders has spent a decade mining the internet for free software treasures. Here's the result of his latest haul...

Lightweight web browser

Dillo 3.0.4

Firefox, Chrome, Internet Explorer, Safari... They're all classed as "web browsers", but a more apt description nowadays would be "application platforms". The amount of code being pushed into browsers for things like 3D graphics, PDF rendering, video codec support and other features is impressive (and sometimes scary) – and we're increasingly running applications and games inside our browsers.

In some ways this is great, but what if you just want a stripped-down web browser? You know, something that renders HTML pages and doesn't try to support everything including the kitchen sink? There are a few options here, and one of our favourites is the classic browser Dillo, which has just seen a new release.

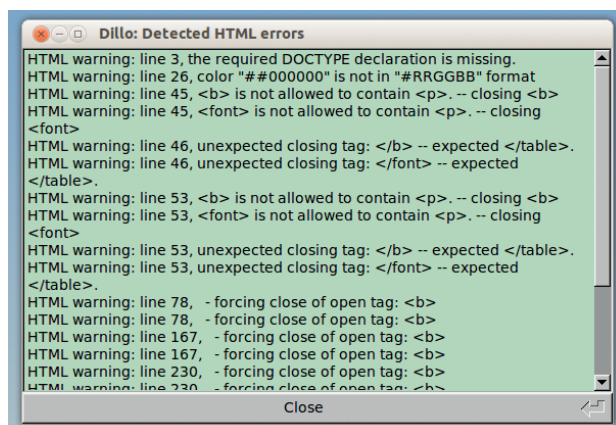
Dillo is designed to be slim and fast, with few dependencies; its interface is built with the frill-free

FLTK toolkit, so once you have the development packages for that installed, you can build it with the usual `./configure`, `make` and `sudo make install` routine.

Back to basics

Dillo doesn't support a lot of web technologies – there's no JavaScript, for instance. That might be a dealbreaker for some people, but we know that plenty of our readers use the NoScript extension to block random JavaScript from being executed on their machines. And many would argue that a site that doesn't work unless JavaScript is enabled is broken by design – sites should really adapt for older or more limited browsers.

"If you do a lot of browsing on text-heavy websites, you'll find Dillo scorchingly fast."

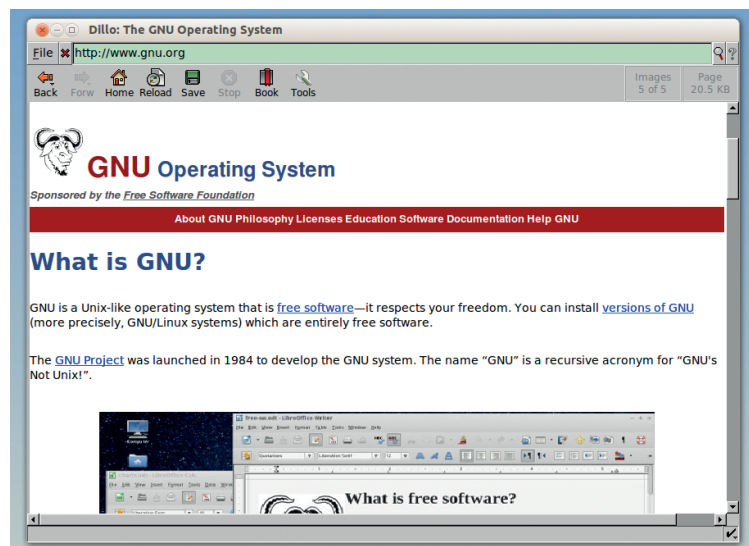


There's also a panel showing errors in a page – useful for checking your HTML before you make it live.

So while Dillo doesn't work everywhere, it still has the basics covered, and handles most HTML and CSS pretty well. If you do a lot of browsing on text-heavy websites, you'll find it scorchingly fast – for instance, loading the Wikipedia page for Linux on Dillo took less than a second in our testing, whereas Firefox took just over three seconds before it finished rendering.

OK, so Firefox does a lot more, but these time savings all add up over an extensive browsing session. And then there's the memory usage: Firefox required 148MB with just that single Wikipedia article open, whereas Dillo only snapped up 20MB from the RAM banks.

Dillo is no challenger to the big browsers, but we love it that there's a browser out there fully focused on being light and fast, getting the most out of older machines.



Dillo's page rendering isn't always perfect, but boy is it fast, and great for reviving old boxes.

PROJECT WEBSITE
www.dillo.org

Task management tool

Taskwarrior 2.3.0

With your work, hobbies, family life and other things, you probably have a sizeable to-do list as well. Installing a task management tool could really help you to get on top of things – it makes all the difference for us. Various programs exist, but we're always happy when we find one that avoids fluffy GUI front-ends and gets down to the heart of the matter.

Taskwarrior is exactly that: it's a supremely configurable command-line program that focuses on managing your to-do list. Entries can be supplemented with priorities, projects, due dates and recurrences (eg the task should be completed once a month). As it's a text-based program, its dependencies are minimal; on our vanilla Ubuntu 13.10 test box we only had to install the **uuid-dev** package in order to follow the instructions in the **INSTALL** file.

Once you have Taskwarrior installed, enter **task** alone – this will create a sample **.taskrc** file in your home directory, along with a **.task/data** directory in the same place. To get started:

```
task add priority:H due:2014-07-10 Write FOSSpicks
```

Here, we're adding a new task with high priority, due on 10 July,

called "Write FOSSpicks". If you then enter **task list**, you'll see the newly created task – try adding a few more tasks with different dates and priorities. When you enter **task list** again you'll see a number at the start of each task; you can use these to perform operations on individual tasks, such as:

```
task 3 delete
```

Groundhog day

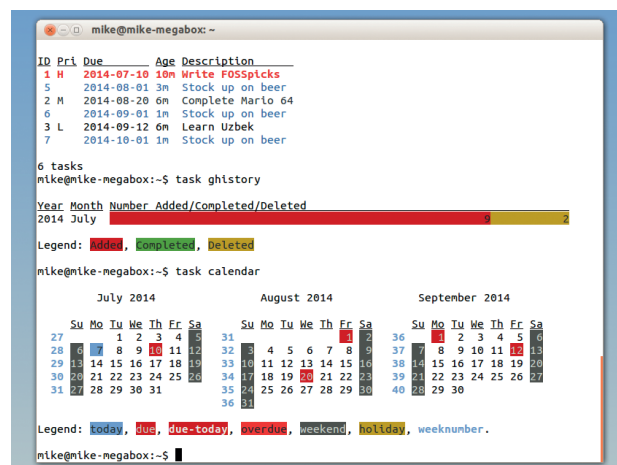
To create a recurring task, specify the date and how often it should happen like so:

```
task add due:1st recur:monthly Stock up on beer
```

This creates a task on the 1st of every month. Now, once you have your to-do list fully inputted, you'll want a way to filter down certain tasks. For instance, you can list only high-priority tasks like so:

```
task priority:H list
```

(Or try **task next** to put the most important tasks first.) Another useful command is **task calendar**, which shows an attractive coloured view of the next few months, with upcoming tasks marked in red. (If



Although it's a CLI app, Taskwarrior makes good use of colour to present information clearly and attractively.

the colour scheme looks rubbish, edit **~/.taskrc** and uncomment one of the theme lines – **light-16.theme** worked best for us.)

As time goes on and you complete more and more of your tasks, you can bring up some other useful views. The **task burndown** command displays a bar chart showing tasks you've started and completed over the previous weeks, and you can change that to a daily view with **task burndown.daily**.

We've only just scratched the surface of Taskwarrior here: enter **man task** to see the vast range of options available.

PROJECT WEBSITE
www.taskwarrior.org

"The 'task burndown' command displays a bar chart showing tasks you've started and completed."

How it works: Adding tasks to projects

```
mike@mike-megabox:~$ task add project:LinuxVoice Write news page
Created task 9.
The project 'LinuxVoice' has changed. Project 'LinuxVoice' is 0%
complete (1 of 1 tasks remaining).
mike@mike-megabox:~$ task list
```

ID	Proj	Pri	Due	Age	Description
1		H	2014-07-10 24m		Write FOSSpicks
5			2014-08-01 17m		Stock up on beer
2		M	2014-08-20 20m		Complete Mario 64
6			2014-09-01 15m		Stock up on beer
3		L	2014-09-12 20m		Learn Uzbek
7			2014-10-01 15m		Stock up on beer
8	LinuxVoice			3s	Write news page

```
7 tasks
mike@mike-megabox:~$
```

```
mike@mike-megabox:~$ task 1 modify project:LinuxVoice
Modifying task 1 'Write FOSSpicks'.
Modified 1 task.
The project 'LinuxVoice' has changed. Project 'LinuxVoice' is 0%
complete (2 of 2 tasks remaining).
mike@mike-megabox:~$ task list
```

ID	Proj	Pri	Due	Age	Description
1	LinuxVoice	H	2014-07-10 25m		Write FOSSpicks
5			2014-08-01 18m		Stock up on beer
2		M	2014-08-20 22m		Complete Mario 64
6			2014-09-01 16m		Stock up on beer
3		L	2014-09-12 22m		Learn Uzbek
7			2014-10-01 16m		Stock up on beer
8	LinuxVoice			1m	Write news page

```
7 tasks
mike@mike-megabox:~$
```

```
mike@mike-megabox:~$ task list project:LinuxVoice
```

ID	Proj	Pri	Due	Age	Description
1	LinuxVoice	H	2014-07-10 27m		Write FOSSpicks
8	LinuxVoice			3m	Write news page

```
2 tasks
mike@mike-megabox:~$ task projects
```

Project	Tasks	Pri:None	Pri:L	Pri:M	Pri:H
(none)	6	4	1	1	0
LinuxVoice	2	1	0	0	1

```
1 project (8 tasks)
mike@mike-megabox:~$
```

1 Create a project

To lump tasks together in projects, first create a task with the **project:Name** option. If that project doesn't yet exist in Taskwarrior, it will be created.

2 Modify tasks

To add existing tasks to projects, use the **modify** command. Here we've done **task 1 modify project:LinuxVoice**, then run **task list**.

3 Filter tasks

To display tasks that belong to a single project, use **task list project:Name**. **task projects** shows all projects in Taskwarrior's database.

Database-less blogging platform

FlatPress 1.0.2

WordPress isn't the hardest thing in the world to install, but sometimes you just want to set up a simple blog or CMS without the hassle of databases and related bits and bobs. FlatPress is designed for exactly these situations: its authors brag that you can "forget about SQL", because all you need is a web server with PHP support. Extract the **.tar.gz** file into your document root, then access **http://localhost/flatpress/** in your browser, and you're ready to go.

Well, almost. FlatPress saves your blog entries as plain text files, so it needs to be able to write these on the server. You're required to make the **fp-content** directory writable by the user account for the web server, but that's pretty much it in terms of fiddling around at the command line.

Next, the installer prompts you to create a login account, and then shows you the front page for your brand-new blog. The default theme isn't especially attractive and has an early-2000s look to it – but there are many extra themes at **http://wiki.flatpress.org/res:themes**, some of which are superb.

FlatPress's admin area is where you add, modify and delete entries, and it also sports an impressive plugin system for adding things like BBCode markup, image thumbnails and spam filtering for comments. There's no fancy WYSIWYG editor for creating entries, just a few buttons providing shortcuts for various formatting options, so it's not designed with complete newbies in mind.

Along with normal entries you can create static pages to appear in the menu. Excellently, you can



To install a new theme, just extract it in **fp-interface/themes/**. This is Deckay40.

modify the layout of the blog (eg where the search box and categories list go) by dragging and dropping widgets – a surprisingly advanced feature for a lightweight engine. And making a backup of the blog or moving it to another server is super-simple, thanks to the lack of a database, as you can simply zip up or copy the directory.

PROJECT WEBSITE
www.flatpress.org

Backup tool

Obnam 1.8

What's the single most important piece of advice you can give to new computer users? "Nigerian princes are not going to send you \$500,000" might come top of the list. Apart from that, the answer is: make backups.

Obnam is a great command line tool for making backups, and has a few nifty tricks up its sleeve. It's easy to build from source, or if you're running a **.deb**-based distro, you can add a repository to get it with a quick **apt-get**. Making a backup is as simple as this:

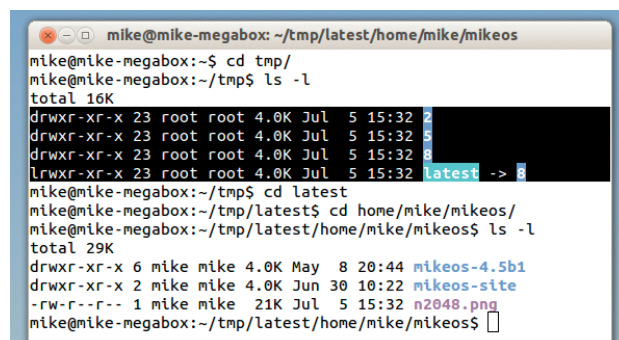
```
obnam backup -r /path/to/backups /dir/to/save
```

Here, Obnam generates a data repository in **/path/to/backups**, saving the contents of **/dir/to/save**. If you run the above command once again, you'll see in the output

that zero bytes are saved. How come? Well, Obnam only saves chunks of data that have changed – so it doesn't waste space by copying every file in each backup. If you make a daily backup of your data but don't add or change many files over a month, the backup repository won't be much bigger than the original. To restore a file:

```
obnam restore --to /restore/to/here /path/to/restore
```

The repository where Obnam stores its backups isn't meant to be human readable. But you can look at generations of backups using the FUSE userspace filesystem driver. Use **cd** to switch into the directory



The highlighted section shows generations of backups, and "latest" is always symlinked to the newest one.

that you've backed up, then enter **mkir tmp**

```
obnam mount --to tmp
```

Now go into **tmp** and you'll see numbered directories: these are different generations of backups. Obnam has commands for removing old generations, leaving, for instance, one per week for the last year behind. It can also back up data via SFTP and supports encryption via GnuPG.

PROJECT WEBSITE
www.obnam.org

"Obnam only changes chunks of data that have changed."

Text editor

NE 2.5

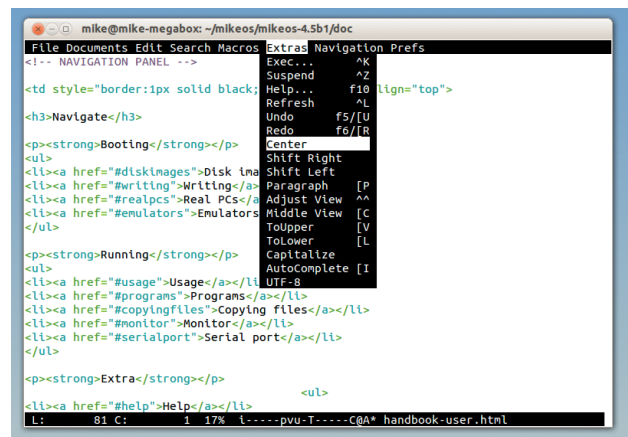
Does the world really need another text editor? We have Vim, Emacs, Nano, Gedit, Kate, Sublime Text, Joe and countless others, covering virtually every type of user on the planet.

Where NE excels, however, is in combining high-end features with newbie-friendly keybindings and menus. It has the approachability of Nano but with much more functionality behind it. Another goal of NE is to be small and highly portable, so it will run on almost every Unix/POSIX-like system (although to be fair, Vi already has that claim to fame).

When you first launch NE, you'll see a status line along the bottom with information about the current file (line and column count, filename and so forth). Hit F1 to bring up the menu – or if your window manager or terminal emulator intercepts F1

for online help, tap Esc twice. You'll see that the menus are chock full of features and options, and many of them have associated keybindings that you can learn for faster editing over time.

NE comes equipped with a boatload of features: syntax highlighting for 46 languages; a macro/scripting system for recording and playing back editing actions; unlimited undo/redo; search and replace with regular expressions; UTF-8 support; and a simple visual file browser. One particularly useful addition for programmers is bracket matching, so you can see if you've properly closed a block of code.



NE's learning curve is much more gradual than Vim's or Emacs's, but it still has oodles of useful features.

In all, NE is a mightily impressive editor. As decade-long Linux geeks we always recommend mastering Vim or Emacs, but we understand that some people simply don't get along with the modal editing or key combinations of those editors. So NE has become our number 3 recommendation now - let us know if you become a fan!

"NE combines high-end features with newbie-friendly keybindings."

PROJECT WEBSITE
<http://ne.di.unimi.it>

Free Flash Player alternative

Lightspark 0.7.2

Picture this: it's the late 2000s. You're Adobe. Your Flash Player is installed on almost every computer around the world, and provides a fairly light and unintrusive way for people to watch videos and play games in their browsers. The HTML5 features that will supersede Flash are a long way off, and you enjoy a very strong position. That was a long time ago.

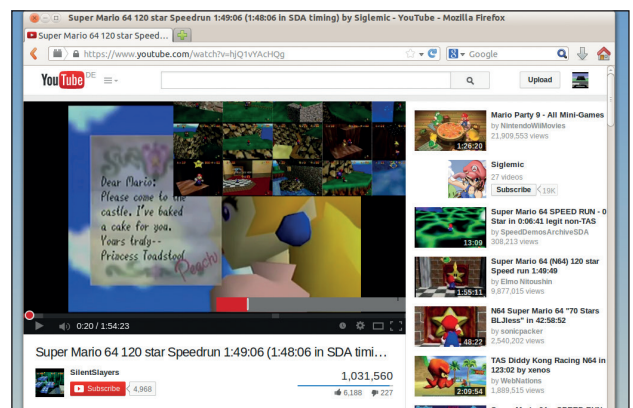
Adobe bloated the player to the extreme – it became increasingly fat and buggy, crashing browsers and generally being a nuisance. And in a shockingly stupid move, Adobe decided to bundle McAfee nagware with the program on Windows, so Flash became universally despised.

Today, most features of Flash are available in the HTML specs, so you can play games and watch many video formats without that ugly

bug-infested binary blob. But many sites still use Flash, and Lightspark is an open source player and browser plugin that aims to provide a replacement for Adobe's program. To compile it you'll need LLVM/Clang – or if you're on an Ubuntu-based distro, a PPA is available.

Free the kittens!

The main focus for Lightspark is YouTube compatibility, and it works fairly well there. We had to edit `/etc/xdg/lightspark.conf` and change the audio backend from "pulseaudio" to "sdl" to get sound, but otherwise most videos played without any major glitches. The main problem we encountered was with jumping around inside videos: trying to get to a later part of a long video was an exercise in frustration, only occasionally working in



Want to jump to a later part of the video? Thumbnails of scenes appear, but moving around didn't work well in our testing.

combination with the pause button. We tried Lightspark with various Flash games and it didn't hold up especially well. But in fairness, it's not at 1.0 yet, and if you want to keep your Linux box pure FOSS but still enjoy the odd YouTube cat compilation video, it bridges the gap until all videos on that site are HTML5-ready.

PROJECT WEBSITE
<http://lightspark.github.io>

CPU reporting tool

i7z 0.27.2

So you've just bought a shiny new desktop or laptop with a high-end Intel Core i7 chip. You've installed your favourite Linux distro and are going about your usual business. Everything is faster than your old box, that's for sure – but don't you want to coo over your snazzy new processor a bit more? Just admire its awesomeness without doing anything specifically productive?

i7z is a CPU reporting tool for these chips, although it also works with i3 and i5 processors. While it's not especially difficult to compile from source code, the developers have made 32-bit and 64-bit static binaries available on the project's website, which work without any hitches (providing you don't mind running pre-compiled binaries).

To get the most out of i7z, run it as root; it gathers various bits of

information about your CPU, and then displays a constantly updating screen. In this screen you'll see general statistics for the processor (frequency, number of cores, whether Hyper-Threading is enabled etc.), and then for each core, the actual frequency at this very moment.

You can also see how much time (as a percentage) each core spends in the various Cx power saving states, along with temperature information. Unlike **top**, there's nothing in the way of interactivity here – the only key combo that does anything is Ctrl+C to exit.

Along with the full-screen mode, there's also a logging option, which

“i7z is a CPU reporting tool for Intel's high-end Core i7 chips.”

```
root@mike-megabox:~# i7z
Cpu speed from cpufreq 2399.00MHz
cpufreq might be wrong if cpufreq is enabled. To guess correctly try estimating via tsc
Linux's inbuilt cpu_khz code emulated now
True Frequency (without accounting Turbo) 2399 MHz
CPU Multiplier 24x || Bus clock frequency (BCLK) 99.96 MHz

Socket [0] - [physical cores=2, logical cores=4, max online cores ever=2]
TURBO ENABLED on 2 Cores, Hyper Threading ON
True Frequency 2498.96 MHz (99.96 x [25])
Max TURBO Multiplier (if Enabled) with 1/2/3/4 Cores is 29x/29x/29x/29x
Current Frequency 799.86 MHz [99.96 x 8.00] (Max of below)
Core [core-id] :Actual Freq (Mult.) C0% Halt(C1)% C3 % C6 % Temp
Core 1 [0]: 799.55 (8.00x) 2.16 97.3 1 1 41
Core 2 [1]: 799.86 (8.00x) 2.66 97.1 1 1 45

C0 = Processor running without halting
C1 = Processor running with halts (States >C0 are power saver)
C3 = Cores running with PLL turned off and core cache turned off
C6 = Everything in C3 + core state saved to last level cache
Above values in table are in percentage over the last 1 sec
[core-id] refers to core-id number in /proc/cpuinfo
'Garbage Values' message printed when garbage values are read
Ctrl+C to exit
```

i7z provides short but handy descriptions of the Cx power saving states along the bottom.

writes the data to a file. This can be useful for performing diagnostics over a longer period of time.

It's also worth noting that the source code includes a Qt-based GUI front end for i7z, but the developers stress that the code is suffering from bit-rot now and it's lacking many features of the command-line version.

PROJECT WEBSITE
<http://code.google.com/p/i7z/>

Bandwidth monitor

Bmon 3.3

As part of the everyday tabs-keeping of an average system administrator (or even just a regular user with more than one machine running), you may have applications chugging away to monitor your disk space and CPU usage. Perhaps they're in a corner of the screen, so you can flick your eyes up and check that the server isn't being overloaded.

If you want to do the same thing for network bandwidth – eg you've put a new web server online and want to make sure it's not getting overstretched – then Bmon is a great solution. It shows bandwidth consumption at a glance, generating simple ASCII graphs that are updated in real time.

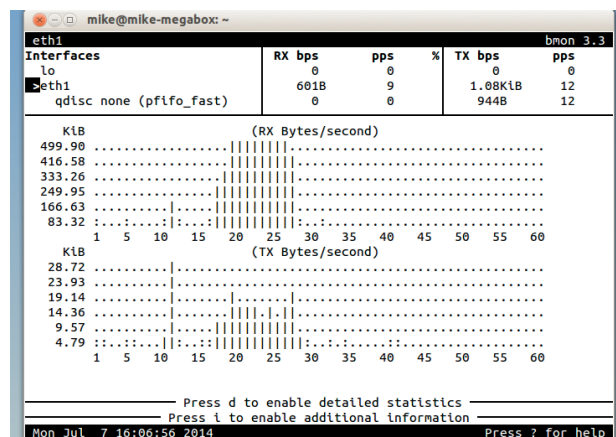
To install it on a Debian/Ubuntu-based distro you'll need the **libnl-3-dev** and **libnl-route-3-dev**

packages installed; then run **./autogen.sh**, **./configure** and **make**. If it compiles successfully, run **make install** as root, and finally **bmon** to start the program.

Network information

Along the top you'll see a list of network interfaces, while the middle panel is where all the action takes place: the auto-updating bar charts (RX = received data, TX = transmitted). Try doing something bandwidth-intensive on the machine, and you'll see the charts update accordingly. This alone is useful enough, making load spikes easy to spot, but if you stretch out the terminal window to provide plenty of space and hit the D key, you'll see lots of additional info.

This is useful for diagnostics, showing things like dropped



Bmon updates its graphs every second by default, but you can change that with a CLI option.

packets and IPv6 statistics. One feature we'd like to see, though, is a warning system: it'd be great if the tool could colour the bar charts in red if the bandwidth exceeds a certain value over a certain period of time, for instance. Time to get cracking on a patch then...

PROJECT WEBSITE
<https://github.com/tgraf/bmon>

FOSSPICKS Brain Relaxers

Physics-happy motobike sim

XMoto 0.5.11

XMoto is one of our favourite open source games of all time, partly because it brings back some fond memories of playing Kickstart 2 on the Speccy, but also because it's simply damn good. At its heart, XMoto is a side-scrolling racing game where your objective is to reach the goal in the quickest time possible – nothing special there.

Where this game is special, though, is in its physics. Keeping your balance is tricky business, especially when you have ramps, slopes and other obstacles to deal with, and trying to land properly off a big jump, when you're spinning through the air, is tough. The up and down cursor keys are used to accelerate and

brake, while the left and right keys shift the rider's weight on the bike, causing him to lean forward or back. This also lets you perform rotations in the air, if you have the guts to do it...

By far the best part of XMoto is its raft of online features. The game can download hundreds of player-created levels from the net, and then you can try to beat the best times. But you're not just racing against the clock: there's a translucent "ghost" rider showing how the record holder completed the track, so you can compare your riding skills with it and end up with some nail-bitingly close finishes.

Some of the levels reward general racing skills, whereas others are downright crazy and require you to perform all manner of



The light blue rider is the "ghost" from the previous record holder - seems a dangerous stunt he's pulling off though...

wacky manoeuvres. It's all superb fun, though, and the vast range of courses to play on will keep you entertained for ages. You can even create your own levels using Inkscape – a great use of another top-quality open source/ Free Software application.

PROJECT WEBSITE
<http://xmoto.tuxfamily.org>

Sliding block puzzle

n2048 0.1

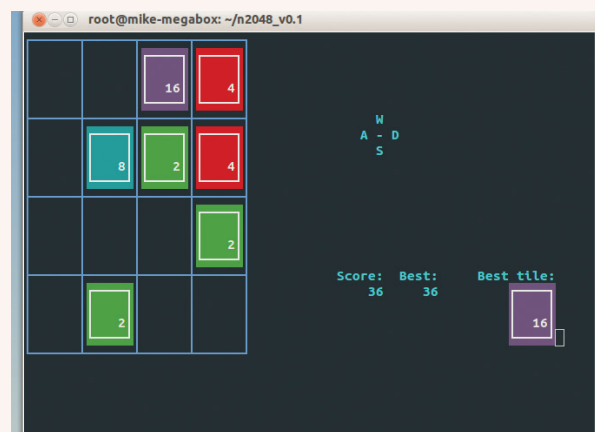
Easy to play, yet endlessly addictive. Tetris is perhaps the best puzzle game in history, but a few others have popped up over the years, and at the moment a game called 2048 is all the rage. It's incredibly easy to pick up: you slide numbered tiles on a grid to combine matching ones, the goal being to eventually create a tile with the number 2048. Sounds easy? It seems so at the start, but it gets darn tough later on.

Anyway, although 2048 is free to play in a web browser (<http://gabrielecirulli.github.io/2048/>), it didn't take long for some open source clones to appear. n2048 runs in a terminal using Ncurses for the interface – so the only

thing you need to compile is the **libncurses5-dev** package (under Debian and Ubuntu; other distros will have similarly named packages). Extract the tarball, enter **make**, and then run the game in place with **./n2048**.

Although the **README.txt** file says that you can use the arrow keys to slide tiles around, they didn't work properly for us – but fortunately you can also use the WASD combination (as in a first-person shooter), or even better, the HJKL keys in true Vi fashion. There's little in the way of fancy graphics or effects here, but the use of colours for differing tile values makes the game easy on the eyes.

n2048 is one of those games you can pick up for a quick session in



As you combine matching tiles to make larger numbers, new tiles (starting with 2) appear on the screen.

your coffee break, yet it's addictive enough to keep you coming back. It also has that Tetris effect where, after an especially long bout of playing, you start seeing tiles when you close your eyes. 📺

PROJECT WEBSITE
www.dettus.net/n2048/